PAML (Phylogenetic Analysis by Maximum Likelihood)

A program package by Ziheng Yang

(Demonstration by Joseph Bielawski)

1. Three inference tasks

(relevant to all approaches)

model based inference

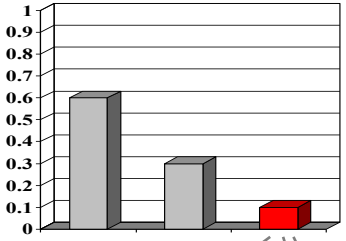
task 1. parameter estimation (e.g., ω)

task 2. hypothesis testing

task 3. make predictions (e.g., sites having $\omega > 1$)

Concept map for tasks 1-3…

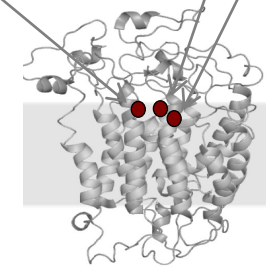
model:
5% have $\omega > 1$



Bayes' rule:
site 4, 12 & 13

```
GTG CTG TCT CCT GCC GAC AAG ACC AAC GTC AAG GCC GCC TGG GGC AAG GTT GGC GCG CAC
... ... ... G.C ... ... ... T.. ..T ... ... ... ... ... ... ... ... .GC A..
... ... ... ..C ..T ... ... ... ... A.. ... A.T ... ... .AA ... A.C ... AGC ...
... ..C ... G.A .AT ... ..A ... ... A.. ... AA. TG. ... ..G ... A.. ..T .GC ..T
... ..C ..G GA. ..T ... ... ..T C.. ..G ..A ... AT. ... ..T ... ..G ..A .GC ...
```

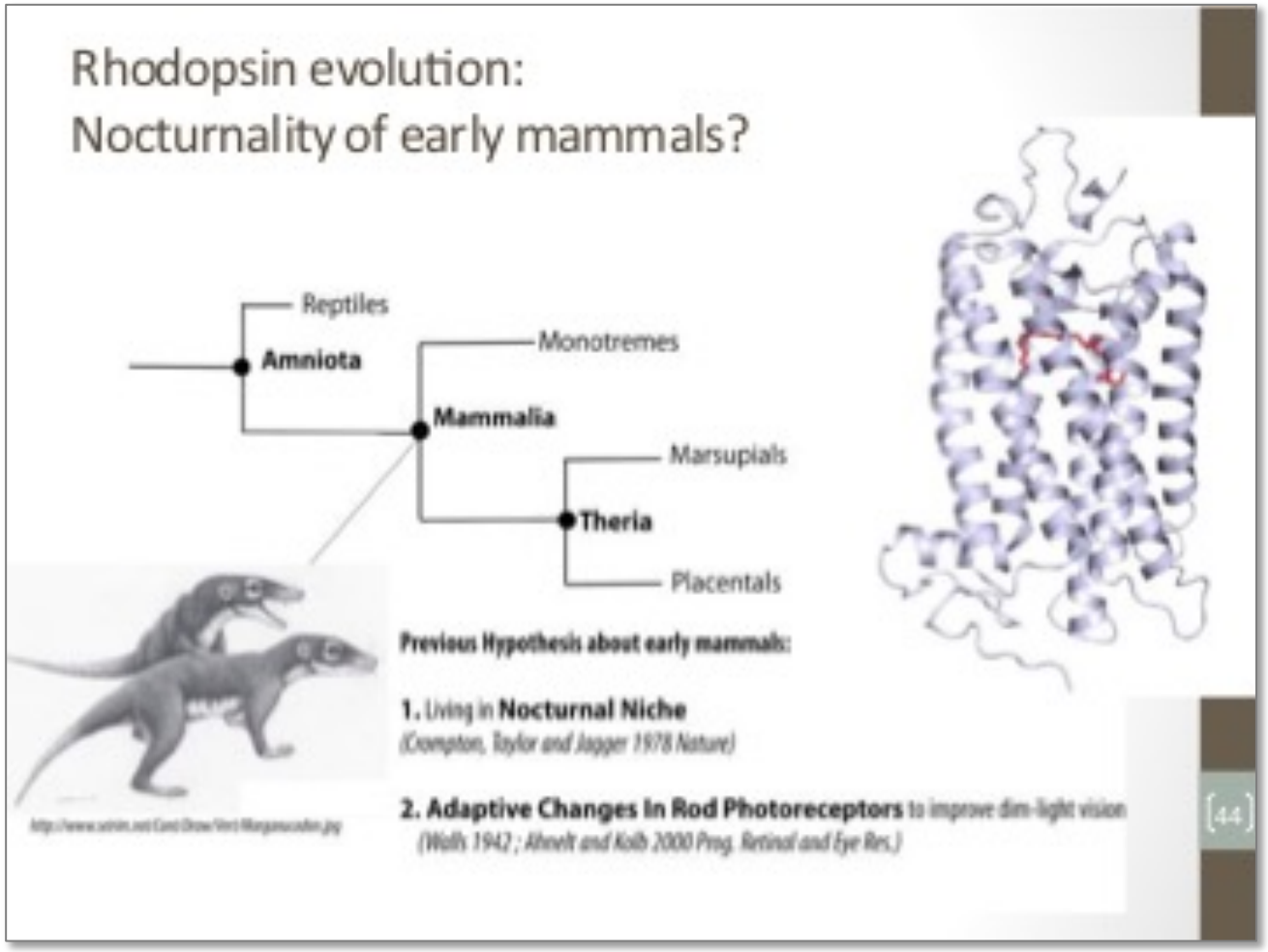
structure:
sites are in contact



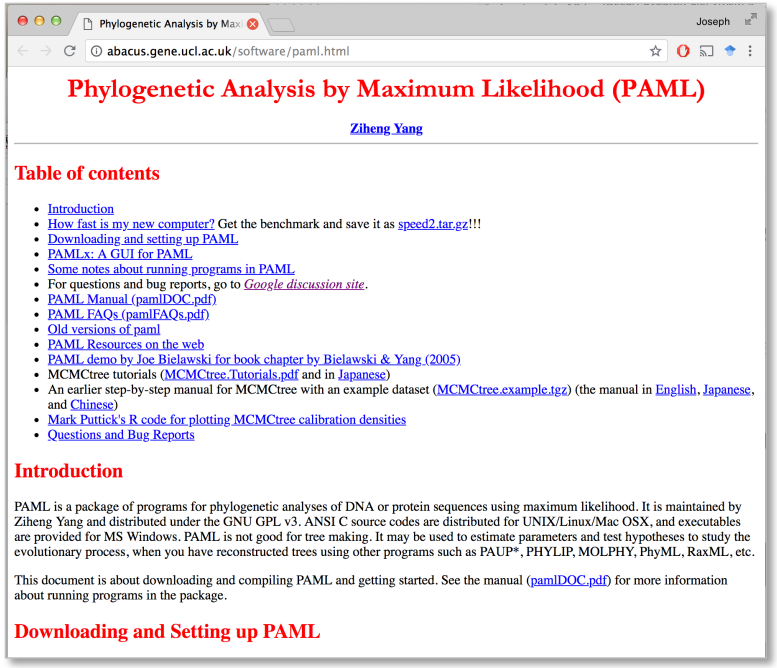
1. Fit model to data → MLEs
2. Test hypotheses via Null and Alternative models for ω
3. Predict which sites have $\omega>1$
4. Interpret results in known biological context

Gold Standard

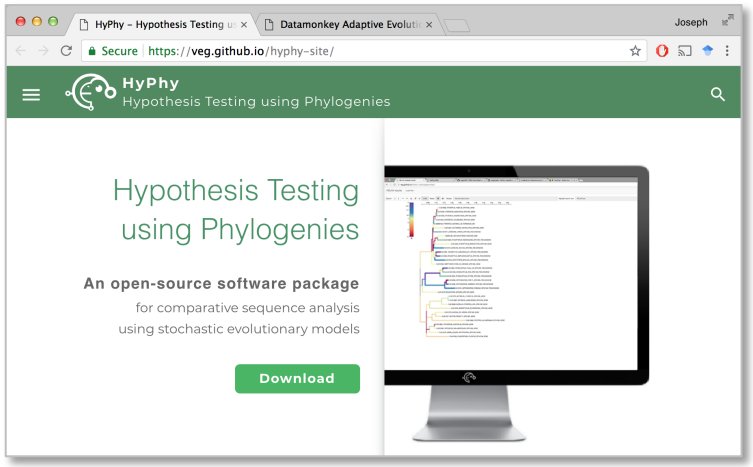
Combine **evolutionary computation with experimental investigation**



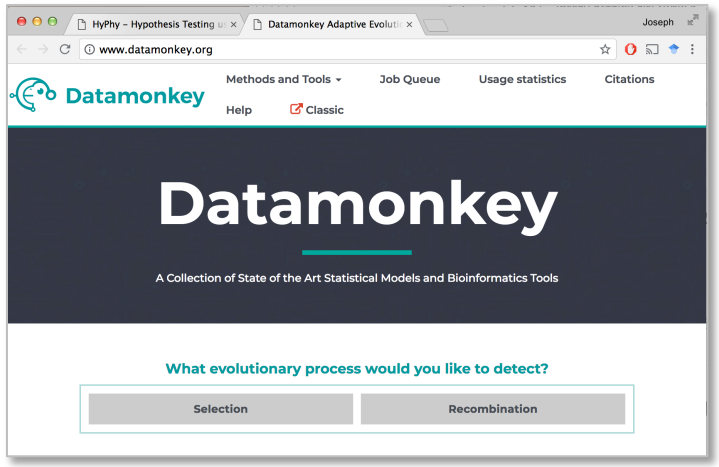
Software: both **PAML** and **HyPhy** are great choices for model-based inference!



http://abacus.gene.ucl.ac.uk/software/paml.html



https://veg.github.io/hyphy-site/



http://www.datamonkey.org/

NOTE: We are actively developing codon models in IQTree (2026)

Objective: To gain a deeper understanding of the basic principles of *model-based inference* in general.

We are NOT tyring to teach a particular software package.

Engauge with the concept questions. It is more important to understand what you are doing (compared to knowing a particular software package).

YOU must attempt to understand the relationship between your model and your data.

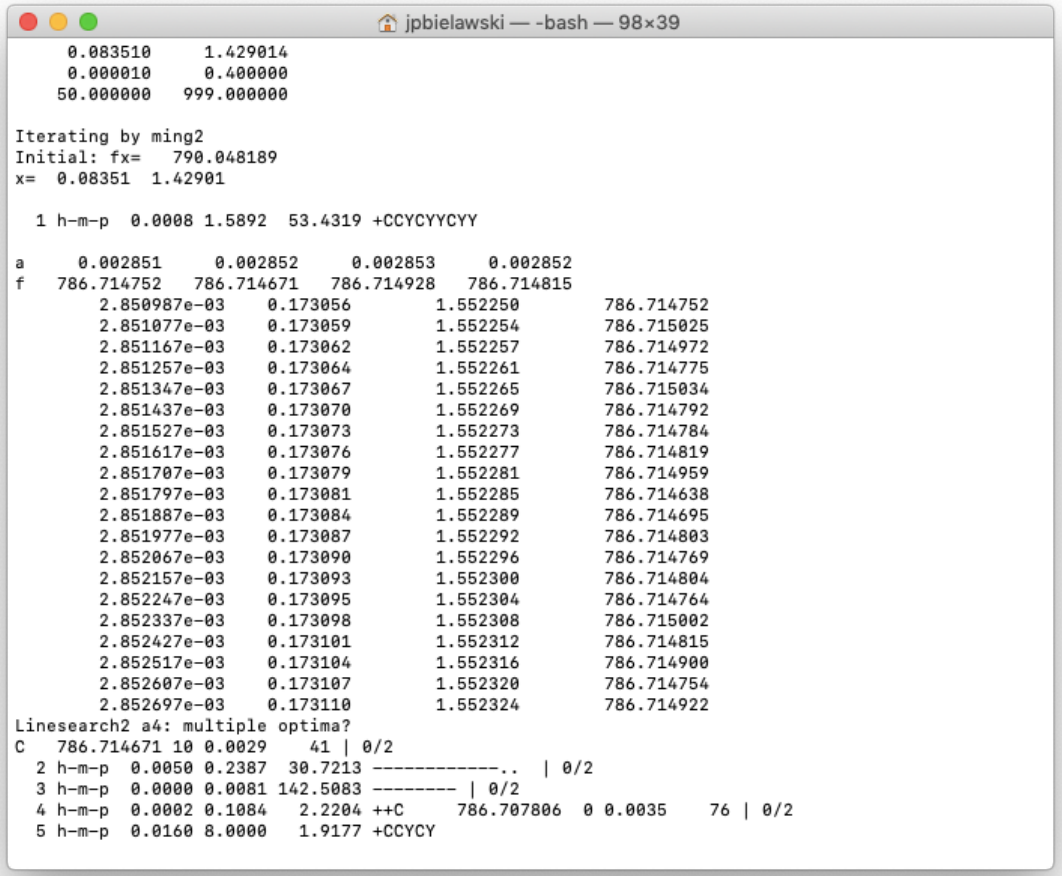
2. Brief introduction to PAML

programs in the package…

baseml	for nucleotide data (bases)
basemlg	continuous-gamma for nucleotides
codeml	**for amino acid & codons data**
evolver	simulation, tree distances
yn00	d_N and d_S by YN00
chi2	chi square table
pamp	parsimony (Yang and Kumar 1996)
mcmctree	Bayes MCMC tree (Yang & Rannala 1997). SLOW

Running PAML programs

1. Sequence data file
2. Tree file
3. Control file (*.ctl)

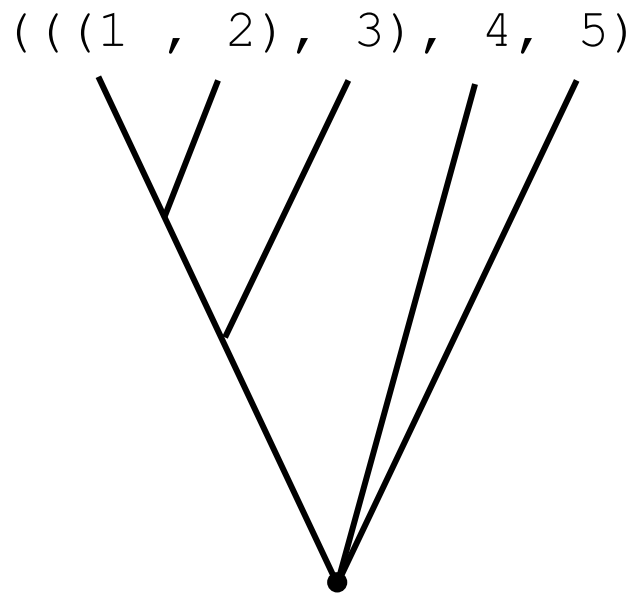


1. sequence file (modified “PHYLIP” format)

```
4  20
sequence_1   TCATT CTATC TATCG TGATG
sequence_2   TCATT CTATC TATCG TGATG
sequence_3   TCATT CTATC TATCG TGATG
sequence_4   TCATT CTATC TATCG TGATG
```

```
4  20
sequence_1TCATTCTATCTATCGTGATG
sequence_2TCATTCTATCTATCGTGATG
sequence_3TCATTCTATCTATCGTGATG
sequence_4TCATTCTATCTATCGTGATG
```

2. tree file (“Newick” format)



This is an **unrooted** tree (basal node is degree = 3)

3. codeml.ctl (the infamous "control file")

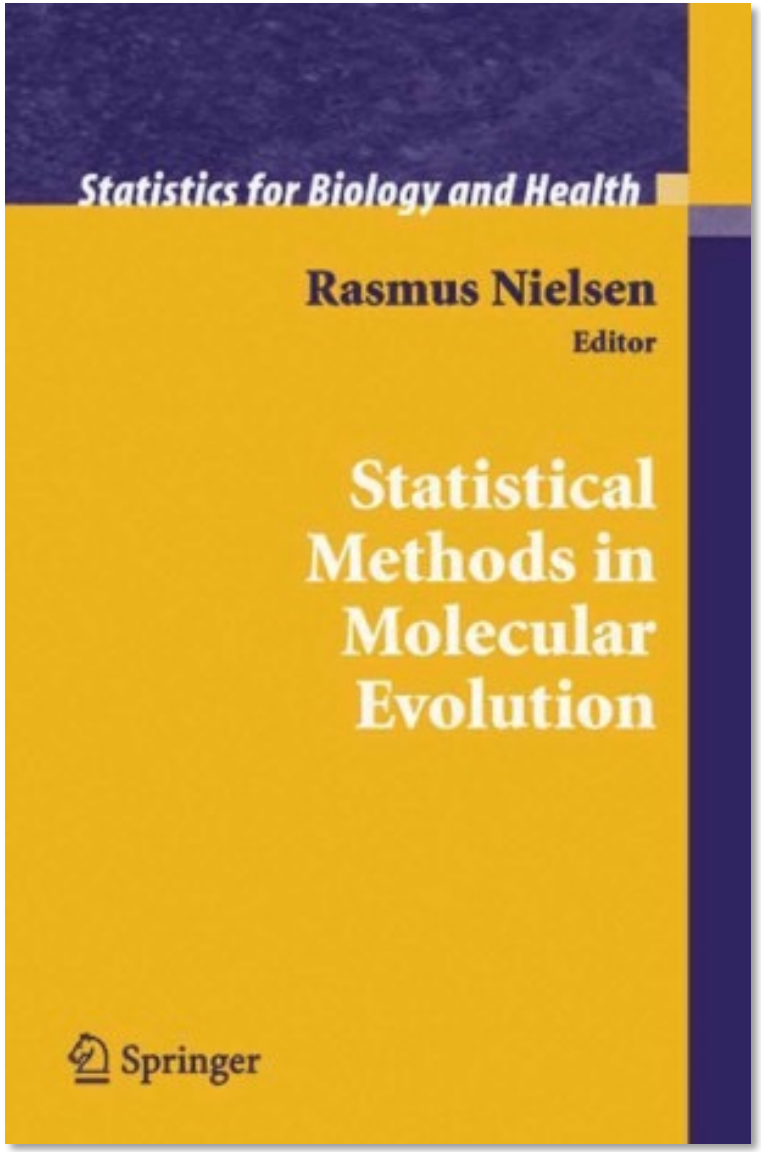
```
  seqfile = seqfile.txt   * sequence data filename
 treefile = tree.txt      * tree structure file name
  outfile = results.txt   * main result file name

    noisy = 9      * 0,1,2,3,9: how much rubbish on the screen
  verbose = 1      * 1:detailed output
  runmode = 0      * 0:user defined tree

  seqtype = 1      * 1:codons
CodonFreq = 2      * 0:equal, 1:F1X4, 2:F3X4, 3:F61

    model = 0      * 0:one omega ratio for all branches

  NSsites = 0      * 0:one omega ratio (M0 in Tables 2 and 4)
                   * 1:neutral (M1 in Tables 2 and 4)
                   * 2:selection (M2 in Tables 2 and 4)
                   * 3:discrete (M3 in Tables 2 and 4)
                   * 7:beta (M7 in Tables 2 and 4)
                   * 8:beta&w; (M8 in Tables 2 and 4)

    icode = 0      * 0:universal code

fix_kappa = 0      * 1:kappa fixed, 0:kappa to be estimated
    kappa = 2      * initial or fixed kappa

fix_omega = 0      * 1:omega fixed, 0:omega to be estimated
    omega = 5      * initial omega

                   *set ncatG for models M3, M7, and M8!!!
   *ncatG = 3      * # of site categories for M3 in Table 4
   *ncatG = 10     * # of site categories for M7 and M8 in Table 4
```

IMPORTANT NOTES:

1. Don't use exercise .ctl files for real data analysis (*they have been modified a little*).

2. Don't use your friends .ctl file for your analysis (*even if he claims it's set up correctly)*

3. The PAML lab



5

Maximum Likelihood Methods for Detecting Adaptive Protein Evolution

Joseph P. Bielawski[1] and Ziheng Yang[2]

[1] Department of Biology, Dalhousie University, Halifax, Nova Scotia B3H 4J1, Canada, j.bielawski@dal.ca
[2] Department of Biology, University College London, Gower Street, London WC1E 6BT, United Kingdom, z.yang@ucl.ac.uk

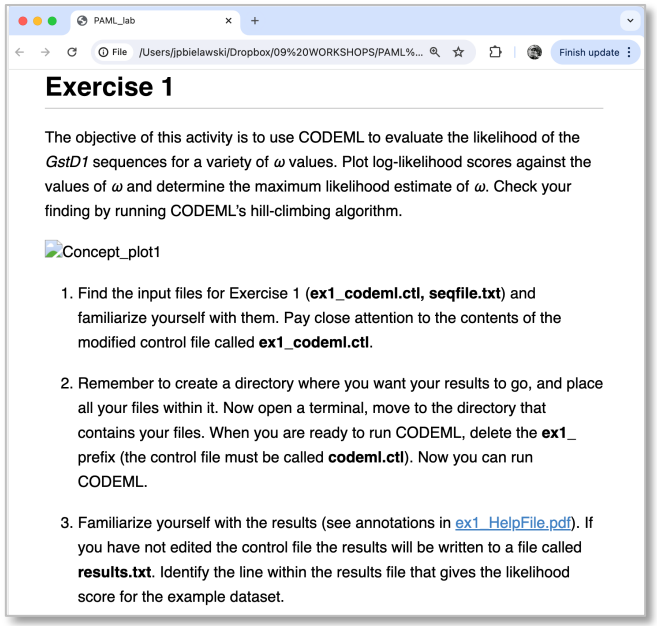
5.1 Introduction

Proteins evolve; the genes encoding them undergo mutation, and the evolutionary fate of the new mutation is determined by random genetic drift as well as purifying or positive (Darwinian) selection. The ability to analyze this process was realized in the late 1970s when techniques to measure genetic variation at the sequence level were developed. The arrival of molecular sequence data also intensified the debate concerning the relative importance of neutral drift and positive selection to the process of molecular evolution [17]. Ever since, there has been considerable interest in documenting cases of molecular adaptation. Despite a spectacular increase in the amount of available nucleotide sequence data since the 1970s, the number of such well-established cases is still relatively small [9, 38]. This is largely due to the difficulty in developing powerful statistical tests for adaptive molecular evolution. Although several powerful tests for nonneutral evolution have been developed [33], significant results under such tests do not necessarily indicate evolution by positive selection.

A powerful approach to detecting molecular evolution by positive selection derives from comparison of the relative rates of synonymous and nonsynonymous substitutions [22]. Synonymous mutations do not change the amino acid sequence; hence their substitution rate (d_S) is neutral with respect to selective pressure on the protein product of a gene. Nonsynonymous mutations do change the amino acid sequence, so their substitution rate (d_N) is a function of selective pressure on the protein. The ratio of these rates ($\omega = d_N/d_S$) is a measure of selective pressure. For example, if nonsynonymous mutations are deleterious, purifying selection will reduce their fixation rate and d_N/d_S will be less than 1, whereas if nonsynonymous mutations are advantageous, they will be fixed at a higher rate than synonymous mutations, and d_N/d_S will be greater than 1. A d_N/d_S ratio equal to one is consistent with neutral evolution.

PAML tutorial translations

English



PAML_lab

/Users/jpbielawski/Dropbox/09%20WORKSHOPS/PAML%...

Finish update

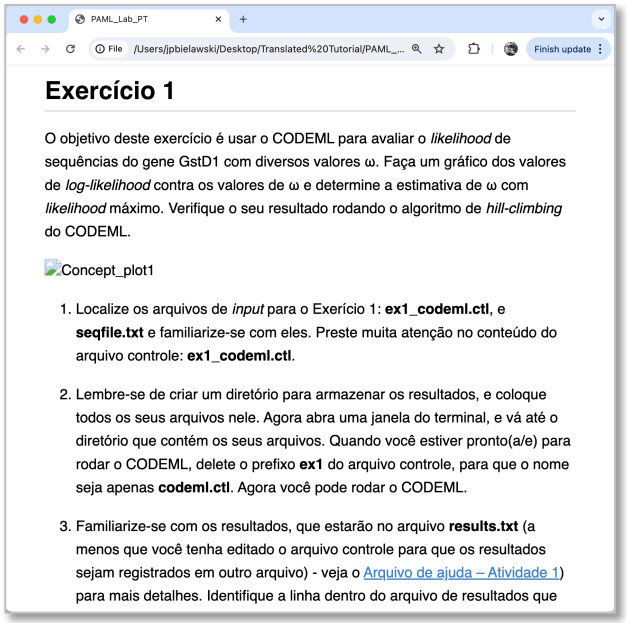
Exercise 1

The objective of this activity is to use CODEML to evaluate the likelihood of the *GstD1* sequences for a variety of *ω* values. Plot log-likelihood scores against the values of *ω* and determine the maximum likelihood estimate of *ω*. Check your finding by running CODEML's hill-climbing algorithm.

Concept_plot1

1. Find the input files for Exercise 1 (**ex1_codeml.ctl, seqfile.txt**) and familiarize yourself with them. Pay close attention to the contents of the modified control file called **ex1_codeml.ctl**.
2. Remember to create a directory where you want your results to go, and place all your files within it. Now open a terminal, move to the directory that contains your files. When you are ready to run CODEML, delete the **ex1_** prefix (the control file must be called **codeml.ctl**). Now you can run CODEML.
3. Familiarize yourself with the results (see annotations in ex1_HelpFile.pdf). If you have not edited the control file the results will be written to a file called **results.txt**. Identify the line within the results file that gives the likelihood score for the example dataset.

Portuguese / Spanish / French



PAML_Lab_PT

/Users/jpbielawski/Desktop/Translated%20Tutorial/PAML_...

Finish update

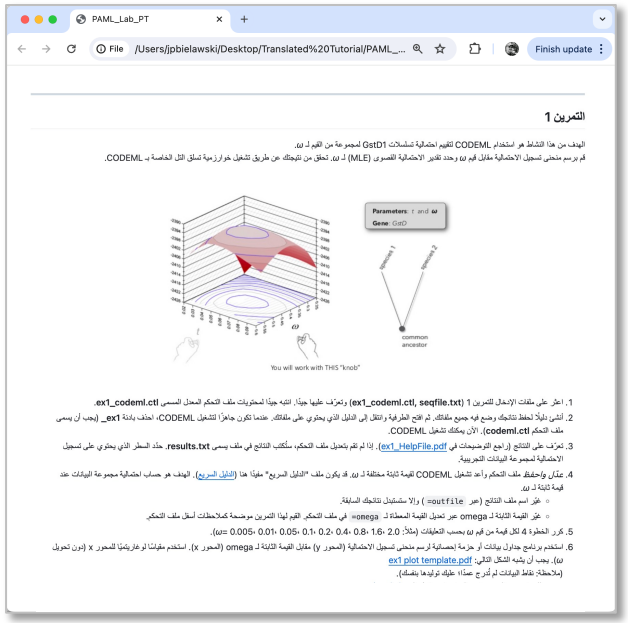
Exercício 1

O objetivo deste exercício é usar o CODEML para avaliar o *likelihood* de sequências do gene GstD1 com diversos valores ω. Faça um gráfico dos valores de *log-likelihood* contra os valores de ω e determine a estimativa de ω com *likelihood* máximo. Verifique o seu resultado rodando o algoritmo de *hill-climbing* do CODEML.

Concept_plot1

1. Localize os arquivos de *input* para o Exercício 1: **ex1_codeml.ctl**, e **seqfile.txt** e familiarize-se com eles. Preste muita atenção no conteúdo do arquivo controle: **ex1_codeml.ctl**.
2. Lembre-se de criar um diretório para armazenar os resultados, e coloque todos os seus arquivos nele. Agora abra uma janela do terminal, e vá até o diretório que contém os seus arquivos. Quando você estiver pronto(a/e) para rodar o CODEML, delete o prefixo **ex1** do arquivo controle, para que o nome seja apenas **codeml.ctl**. Agora você pode rodar o CODEML.
3. Familiarize-se com os resultados, que estarão no arquivo **results.txt** (a menos que você tenha editado o arquivo controle para que os resultados sejam registrados em outro arquivo) - veja o Arquivo de ajuda – Atividade 1) para mais detalhes. Identifique a linha dentro do arquivo de resultados que

Arabic



PAML_Lab_PT

/Users/jpbielawski/Desktop/Translated%20Tutorial/PAML_...

Finish update

التمرين 1

الهدف من هذا النشاط هو استخدام CODEML لتقييم احتمالية تسلسلات GstD1 لمجموعة من القيم لـ ω.

قم برسم منحنى تسجيل الاحتمالية مقابل قيم ω وحدد تقدير الاحتمالية القصوى (MLE) لـ ω. تحقق من نتيجتك عن طريق تشغيل خوارزمية تسلق التل الخاصة بـ CODEML.

1. اعثر على ملفات الإدخال للتمرين 1 (**ex1_codeml.ctl, seqfile.txt**) وتعرّف عليها جيدًا. انتبه جيدًا لمحتويات ملف التحكم المعدل المسمى **ex1_codeml.ctl**.
2. أنشئ دليلًا لحفظ نتائجك وضع فيه جميع ملفاتك. ثم افتح الطرفية وانتقل إلى الدليل الذي يحتوي على ملفاتك. عندما تكون جاهزًا لتشغيل CODEML، احذف بادئة **ex1_** (يجب أن يسمى ملف التحكم **codeml.ctl**). الآن يمكنك تشغيل CODEML.
3. تعرّف على النتائج (راجع التوضيحات في ex1_HelpFile.pdf). إذا لم تقم بتعديل ملف التحكم، ستُكتب النتائج في ملف يسمى **results.txt**. حدّد السطر الذي يحتوي على تسجيل الاحتمالية لمجموعة البيانات التجريبية.
4. عدّل واحفظ ملف التحكم وأعد تشغيل CODEML لقيمة ثابتة مختلفة لـ ω. قد يكون ملف "الدليل السريع" مفيدًا هنا (الدليل السريع). الهدف هو حساب احتمالية مجموعة البيانات عند قيمة ثابتة لـ ω.
   - غيّر اسم ملف النتائج (عبر `=outfile`) وإلا ستستبدل نتائجك السابقة.
   - غيّر القيمة الثابتة لـ omega عبر تعديل القيمة المعطاة لـ `=omega` في ملف التحكم. القيم موضحة كملاحظات أسفل ملف التحكم.
5. كرر الخطوة 4 لكل قيمة من قيم ω بحسب التعليقات (مثلًا: 2.0، 1.6، 0.8، 0.4، 0.2، 0.1، 0.05، 0.01، 0.005= ω).
6. استخدم برنامج جداول بيانات أو حزمة إحصائية لرسم منحنى تسجيل الاحتمالية (المحور y) مقابل القيمة الثابتة لـ omega (المحور x). استخدم مقياسًا لوغاريتميًا للمحور x (دون تحويل ω). يجب أن يشبه الشكل التالي: ex1 plot template.pdf
(ملاحظة: نقاط البيانات لم تُدرج عمدًا؛ عليك توليدها بنفسك).

Bioinformatics-4-Everyone (B4B) github: https://github.com/Evolution-for-Everyone/bioinformatics-for-everyone

The Evolution-4-Everyone (E4B) Project: https://evolution-for-everyone.github.io/evolution4everyone/

Accessing the files

1. If you are doing the lab AT THE WORKSHOP: On the virtual machines we will be using in the 2022 workshop, there will be a symlink in your home directory named "moledata" that takes you to the course data files. There you will find directories for the various labs (e.g., MSAlab, revbayes, PamlLab, etc.).

To view the list of labs type:

```
ls ~/moledata
```

To view the contents of the Paml Lab type:

```
ls -l ~/moledata/PamlLab
```

This will reveal the directories for each excercise:

```
ex1
ex2
ex3
ex4
```

The files are already on the virtual machine you are using. *However, you will want to run each exercise in a separate directory that you will create*. So, create a new directory. The name of the new directory is up to you, but pick something informative (e.g., ~/PAML_ex1).

To *copy* the files required for exercise 1 just type:

```
cp ~/moledata/PamlLab/ex1/* ~/PAML_ex1
```

Now you are ready to do exercise 1 within `~/PAML_ex1`

This is STEP 1

2. If you are doing the lab INDEPNDENTLY of the workshop: You can do this lab by downloading all the necessary files from an archive here, or you can download the files individually for each exercise as you need them here.

Either way, it is still *"best practice" to run each exercise in a separate working directory that you will create* (*e.g.*, `PAML_ex1`), *and work with copies of the required files within that directory*.

Re-naming files: 2 important points…

1. For each exercise you must **remove the "`exN_`" prefix** from the control files

   for example: **`cp ex1_codeml.ctl codeml.ctl`**

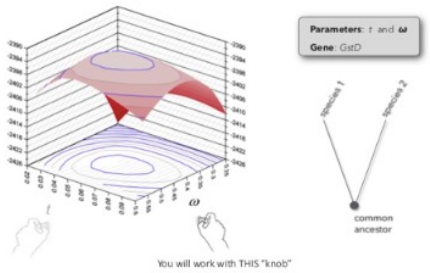
2. PAML will overwrite its own outfiles without warning you!!! **Rename any outfiles you want to save!!!**

Step-by-step protocols

awarnach.mathstat.dal.ca

Exercise 1

The objective of this activity is to use CODEML to evaluate the likelihood of the *GstD1* sequences for a variety of ω values. Plot log-likelihood scores against the values of ω and determine the maximum likelihood estimate of ω. Check your finding by running CODEML's hill-climbing algorithm.



1. Find the input files for Exercise 1 (**ex1_codeml.ctl, seqfile.txt**) and familiarize yourself with them. Pay close attention to the contents of the modified control file called **ex1_codeml.ctl**.
2. Remember to create a directory where you want your results to go, and place all your files within it. Now open a terminal, move to the directory that contains your files. When you are ready to run CODEML, delete the **ex1**_ prefix (the control file must be called **codeml.ctl**). Now you can run CODEML.
3. Familiarize yourself with the results (see annotations in ex1_HelpFile.pdf). If you have not edited the control file the results will be written to a file called **results.txt**. Identify the line within the results file that gives the likelihood score for the example dataset.
4. Now *change and save* the control file and re-run CODEML for a different fixed value of ω. The control file "quick guide" might be helpful here (quick guide). The objective is to compute the likelihood of the example dataset given a fixed value of ω. *Change the control file as follows*:
   - Change the name of your result file (via `outfile=` in the control file) or you will overwrite your previous results!

results "help-files"

Exercise 1 help file: This file contains an annotated portion of the results output by codeml for a maximum likelihood analysis of a pair of sequences. The box contains the portion of the results file that is most relevant to completing exercise 1. These lines of the output can be found at the end of the results file.

```
.
.
.

pairwise comparison, codon frequencies: Fcodon.


2 (Sim) ... 1 (Mel)
lnL = -786.354023
  0.17748  2.24589

t= 0.1775  S=    44.6  N=   555.4  dN/dS= 0.0010  dN= 0.0008  dS= 0.7866
```

This line indicates a pairwise comparison. "Sim" and "Mel" are the sequence labels provided in the sequence file. 1 and 2 indicate the order of these sequences in that file.

This line gives the log likelihood (ln L) of the pair of sequences

This is the value of ω. In this case it was fixed = 0.001

In 2025…

- **exercises 1, 3 & 4** we will do together (ex 4, step 8 is optional)

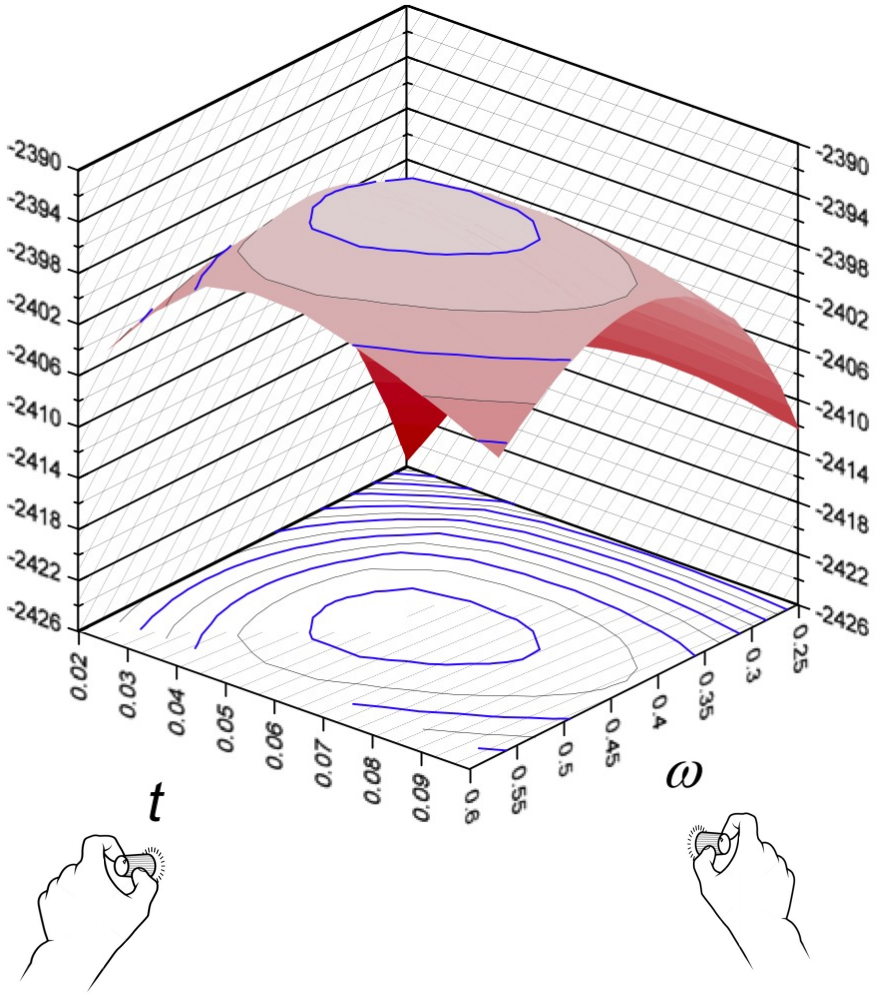
- we will SKIP exercise 2

Work in teams and discuss your progress!!!

Exercise 1:

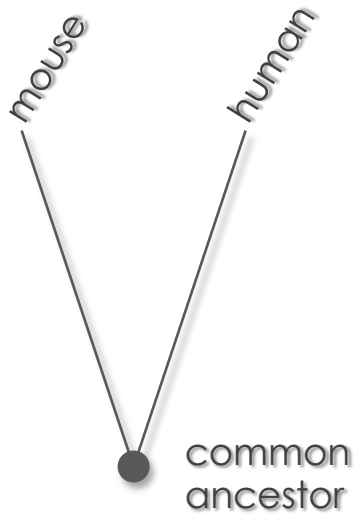
ML estimation of the d_N/d_S (ω) "by hand" for *GstD1*

exercise 1:



Parameters: t and ω

Gene: acetylcholine α receptor



lnL = -2399

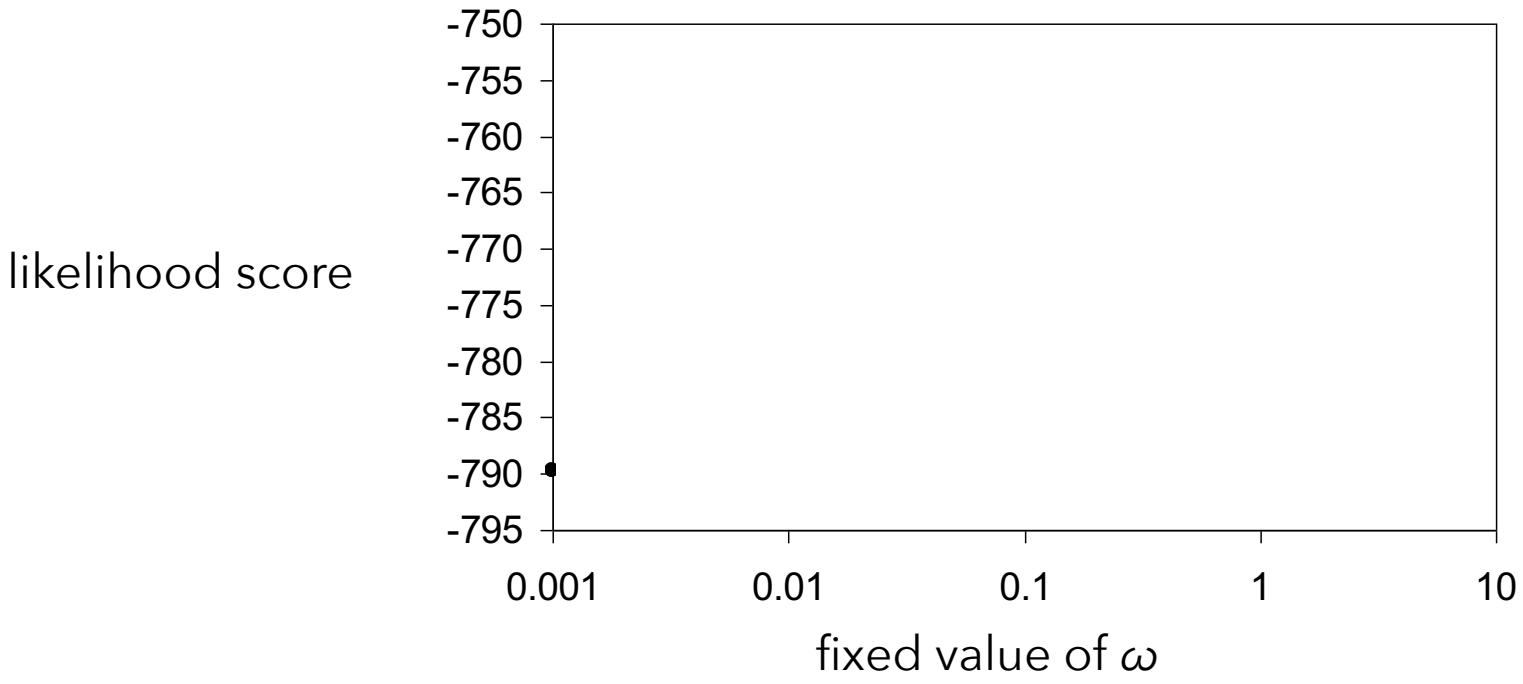
exercise 1:
you will work THIS "knob"

exercise 1:

```
  seqfile = seqfile.txt          * sequence data filename
  outfile = results_0.001.txt    * main result file name [CHANGE THIS]

    noisy = 9      * 0,1,2,3,9: how much rubbish on the screen
  verbose = 1      * 1:detailed output
  runmode = -2     * -2:pairwise

  seqtype = 1      * 1:codons
CodonFreq = 3      * 0:equal, 1:F1X4, 2:F3X4, 3:F61
    model = 0      *
  NSsites = 0      *
    icode = 0      * 0:universal code

fix_kappa = 0      * 1:kappa fixed, 0:kappa to be estimated
    kappa = 2      * initial or fixed kappa

fix_omega = 1      * 1:omega fixed, 0:omega to be estimated
    omega = 0.001  * 1st fixed omega value  [CHANGE THIS]


   *NOTEs: alternate fixed omega values
   *omega = 0.005  * 2nd fixed value
   *omega = 0.01   * 3rd fixed value
   *omega = 0.05   * 4th fixed value
   *omega = 0.10   * 5th fixed value
   *omega = 0.20   * 6th fixed value
   *omega = 0.40   * 7th fixed value
   *omega = 0.80   * 8th fixed value
   *omega = 1.60   * 9th fixed value
   *omega = 2.00   * 10th fixed value
```

exercise 1:

plot: likelihood score vs. omega (log scale)



exercise 1:

```
  seqfile = seqfile.txt          * sequence data filename
  outfile = results_0.001.txt    * main result file name [CHANGE THIS]

    noisy = 9       * 0,1,2,3,9: how much rubbish on the screen
  verbose = 1       * 1:detailed output
  runmode = -2      * -2:pairwise

  seqtype = 1       * 1:codons
CodonFreq = 3       * 0:equal, 1:F1X4, 2:F3X4, 3:F61
    model = 0       *
  NSsites = 0       *
    icode = 0       * 0:universal code

fix_kappa = 0       * 1:kappa fixed, 0:kappa to be estimated
    kappa = 2       * initial or fixed kappa

fix_omega = 1       * 1:omega fixed, 0:omega to be estimated
    omega = 0.001   * 1st fixed omega value  [CHANGE THIS]

  *NOTEs: alternate fixed omega values
  *omega = 0.005   * 2nd fixed value
  *omega = 0.01    * 3rd fixed value
  *omega = 0.05    * 4th fixed value
  *omega = 0.10    * 5th fixed value
  *omega = 0.20    * 6th fixed value
  *omega = 0.40    * 7th fixed value
  *omega = 0.80    * 8th fixed value
  *omega = 1.60    * 9th fixed value
  *omega = 2.00    * 10th fixed value
```

When you are done...

set...

```
fix_omega = 0
omega = 10
```

... now codeml will estimate the MLE for omega

exercise 1 concept questions:

1. How close was your “by-hand” estimate of the MLE compared to the one produced by the codeml optimization algorithm?

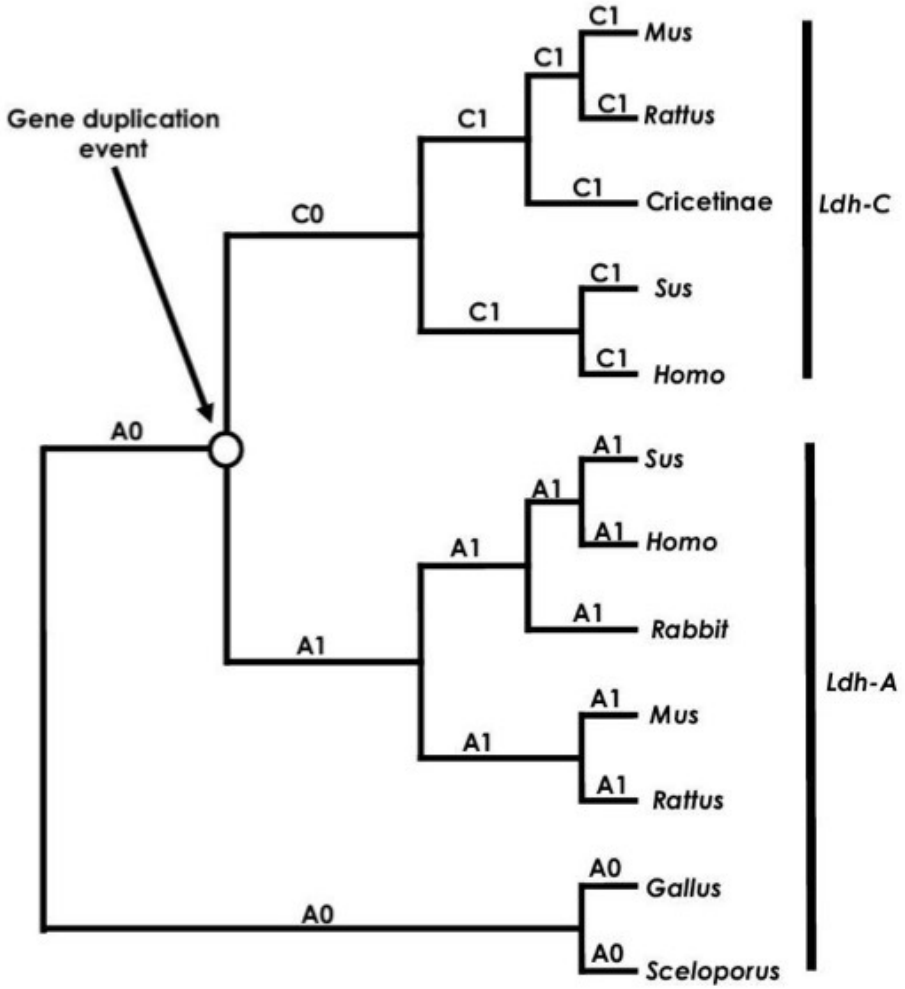
2. Does the area under your likelihood curve sum to 1.0?

3. Can you explain*, in non-technical language*, what the MLE represents and why you would want to estimate it?

Exercise 3:

Test hypotheses about molecular evolution of *Ldh* gene family

exercise 3:



Each one represents a different “branch model”

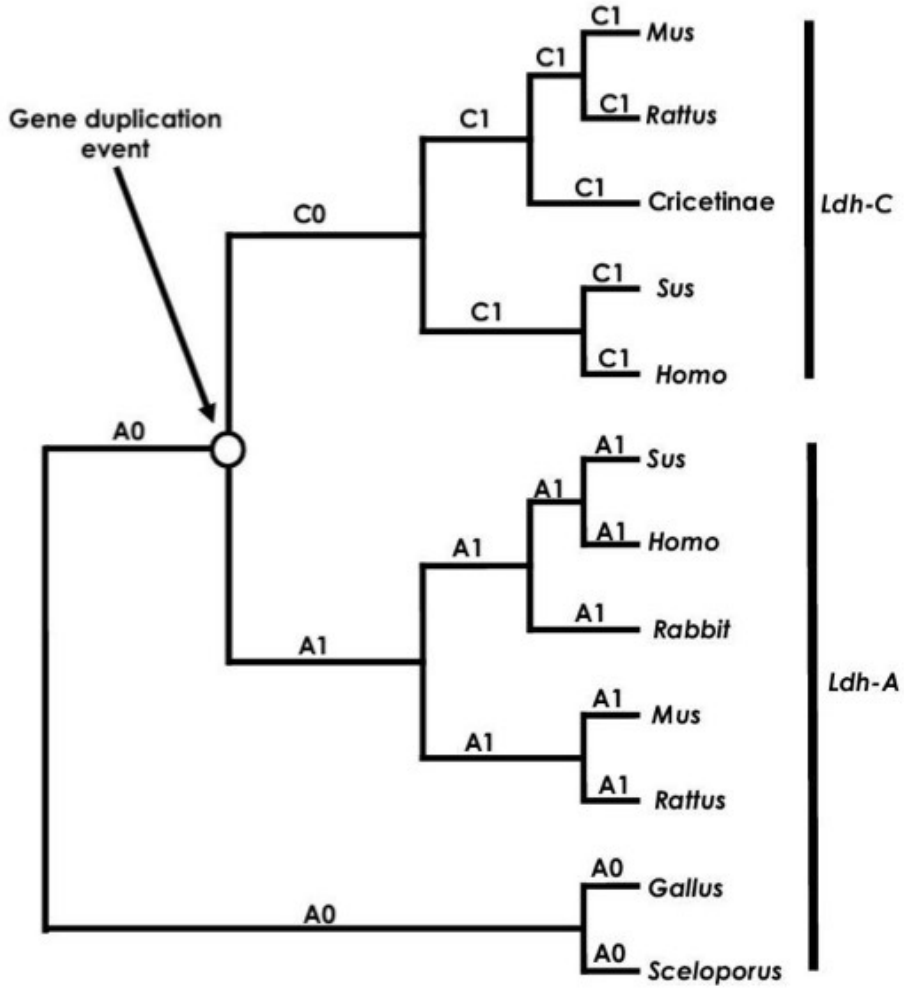
H_0: $\omega_{A0} = \omega_{A1} = \omega_{C1} = \omega_{C0}$

H_1: $\omega_{A0} = \omega_{A1} = \omega_{C1} \neq \omega_{C0}$

H_2: $\omega_{A0} = \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

H_3: $\omega_{A0} \neq \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

exercise 3:



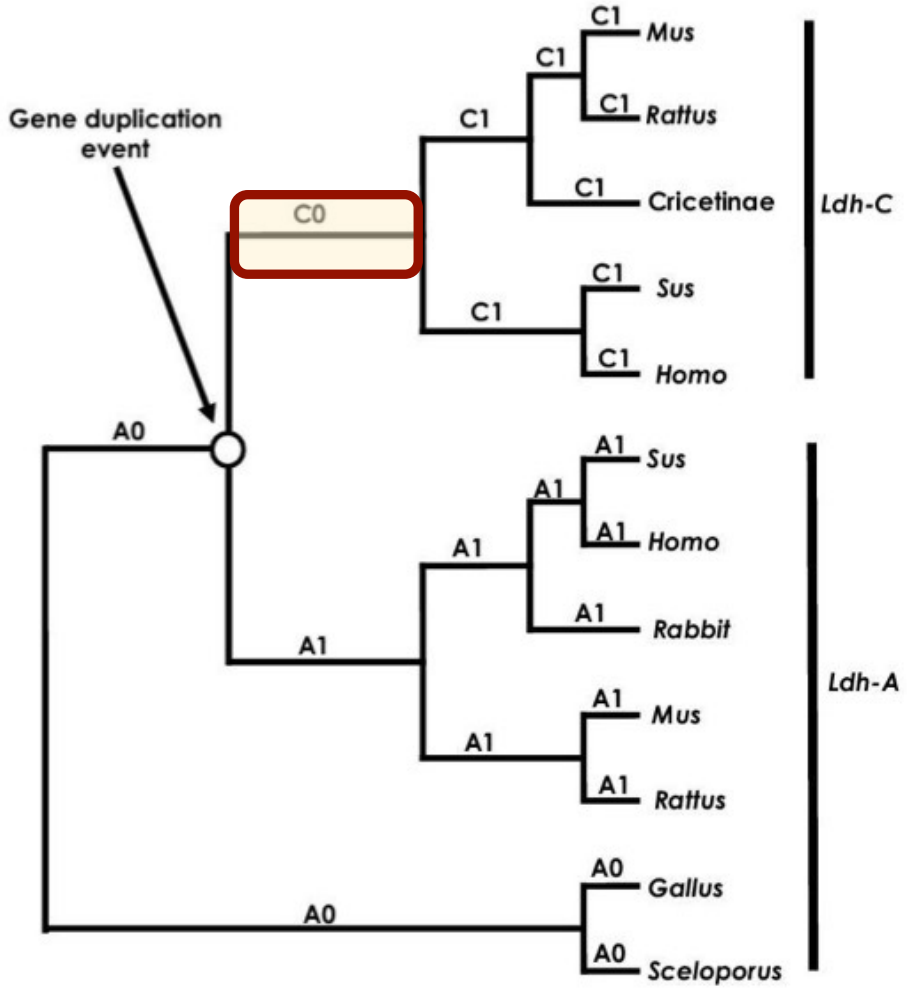
H_0: $\omega_{A0} = \omega_{A1} = \omega_{C1} = \omega_{C0}$ Null model

H_1: $\omega_{A0} = \omega_{A1} = \omega_{C1} \neq \omega_{C0}$

H_2: $\omega_{A0} = \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

H_3: $\omega_{A0} \neq \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

exercise 3:



H_0: $\omega_{A0} = \omega_{A1} = \omega_{C1} = \omega_{C0}$

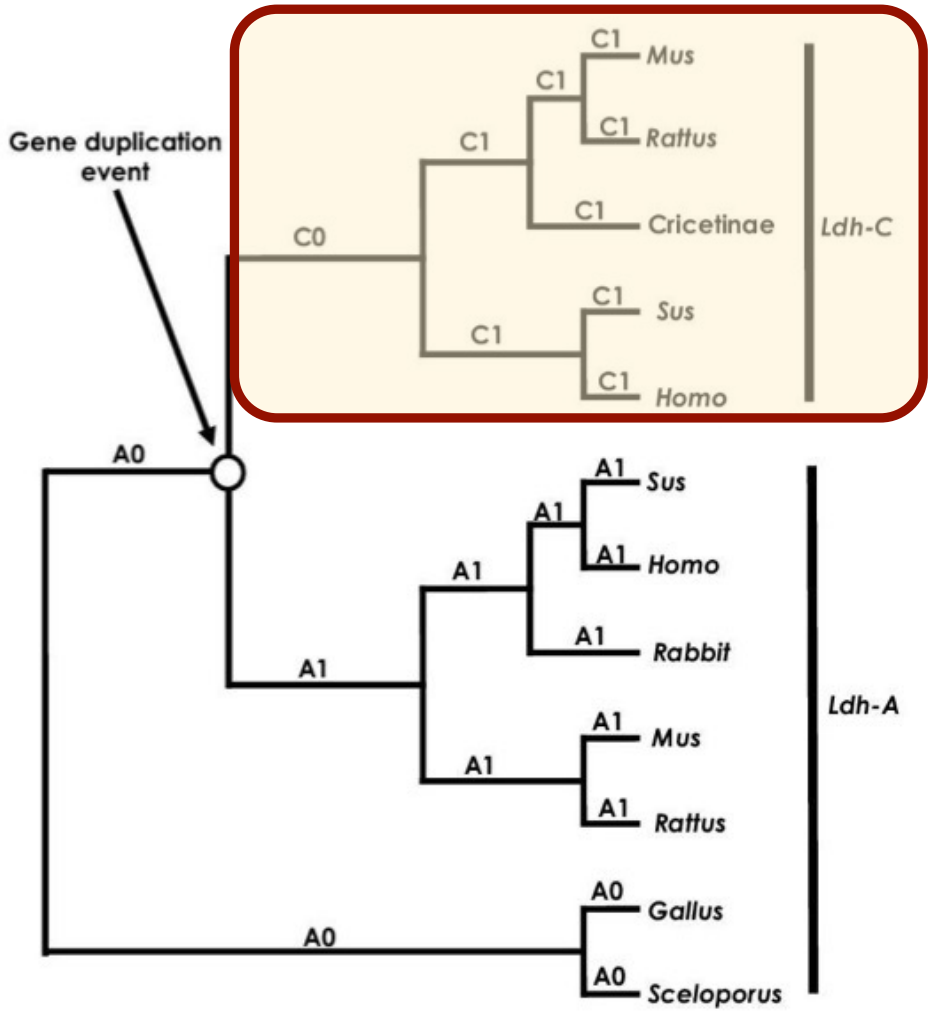
H_1: $\omega_{A0} = \omega_{A1} = \omega_{C1} \neq \omega_{C0}$

H_2: $\omega_{A0} = \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

H_3: $\omega_{A0} \neq \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

Episodic model

exercise 3:



H_0: $\omega_{A0} = \omega_{A1} = \omega_{C1} = \omega_{C0}$

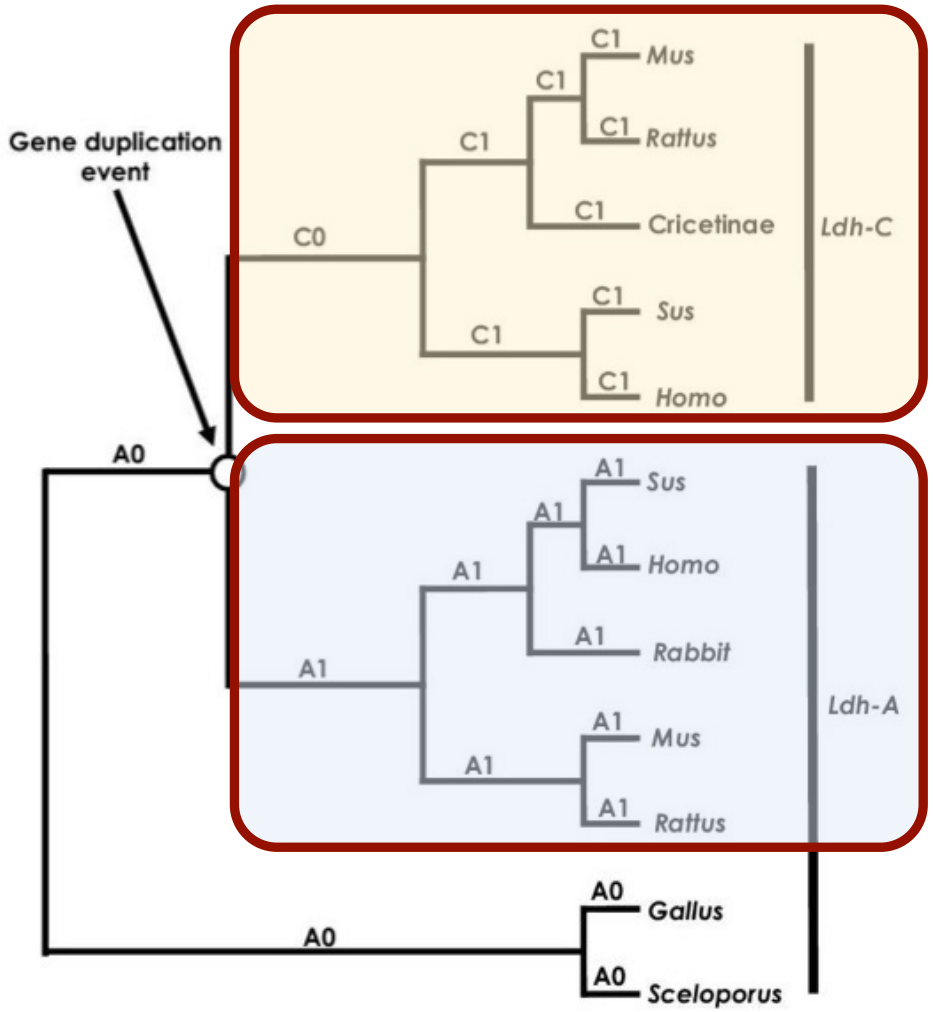
H_1: $\omega_{A0} = \omega_{A1} = \omega_{C1} \neq \omega_{C0}$

H_2: $\omega_{A0} = \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

H_3: $\omega_{A0} \neq \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

Long-term shift: 1-clade model

exercise 3:



H_0: $\omega_{A0} = \omega_{A1} = \omega_{C1} = \omega_{C0}$

H_1: $\omega_{A0} = \omega_{A1} = \omega_{C1} \neq \omega_{C0}$

H_2: $\omega_{A0} = \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

H_3: $\omega_{A0} \neq \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

Long-term shift: 2-clade model

exercise 3:

```
      seqfile = seqfile.txt   * sequence data filename
     treefile = tree.H0.txt   * tree structure file name [CHANGE THIS]
      outfile = results.txt   * main result file name

        noisy = 9      * 0,1,2,3,9: how much rubbish on the screen
      verbose = 1      * 1:detailed output
      runmode = 0      * 0:user defined tree

      seqtype = 1      * 1:codons
    CodonFreq = 2      * 0:equal, 1:F1X4, 2:F3X4, 3:F61

        model = 0      * 0:one omega ratio for all branches  [FOR MODEL H0]
                       * 1:separate omega for each branch
                       * 2:user specified dN/dS ratios for branches [FOR MODELS H1-H3]

      NSsites = 0      *

        icode = 0      * 0:universal code

    fix_kappa = 0      * 1:kappa fixed, 0:kappa to be estimated
        kappa = 2      * initial or fixed kappa

    fix_omega = 0      * 1:omega fixed, 0:omega to be estimated
        omega = 0.2    * initial omega
```

*H_0 in Table 3:
*model = 0
*(X02152Hom,U07178Sus,(M22585rab,((NM017025Rat,U13687Mus),
*(((AF070995C,(X04752Mus,U07177Rat)),(U95378Sus,U13680Hom)),(X53828OG1,
* U28410OG2)))));

*H_1 in Table 3:
*model = 2
*(X02152Hom,U07178Sus,(M22585rab,((NM017025Rat,U13687Mus),(((AF070995C,
*(X04752Mus,U07177Rat)),(U95378Sus,U13680Hom))#1,(X53828OG1,U28410OG2))
*)));

*H_2 in Table 3:
*model = 2
* (X02152Hom,U07178Sus,(M22585rab,((NM017025Rat,U13687Mus),(((AF070995C
* #1,(X04752Mus #1,U07177Rat #1)#1)#1,(U95378Sus #1,U13680Hom #1)
* #1)#1,(X53828OG1,U28410OG2)))));

*H_3 in Table 3:
*model = 2
* (X02152Hom,U07178Sus,(M22585rab,((NM017025Rat,U13687Mus),(((AF070995C
* #1,(X04752Mus #1,U07177Rat #1)#1)#1,(U95378Sus #1,U13680Hom #1)
* #1)#1,(X53828OG1 #2,U28410OG2 #2)#2))));

Null model

Episodic model

Long-term shift: 1-clade model

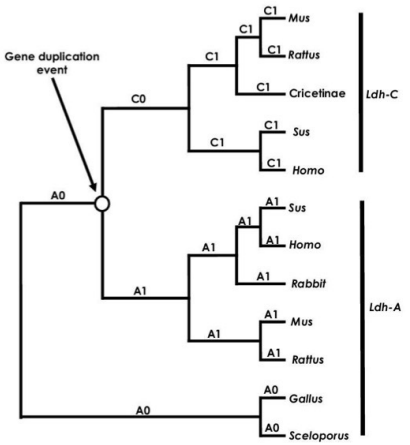
Long-term shift: 2-clade model

exercise 3:

```
  seqfile = seqfile.txt   * sequence data filename
 treefile = tree.H0.txt   * tree structure file name [CHANGE THIS]
  outfile = results.txt   * main result file name

    noisy = 9      * 0,1,2,3,9: how much rubbish on the screen
  verbose = 1      * 1:detailed output
  runmode = 0      * 0:user defined tree

  seqtype = 1      * 1:codons
CodonFreq = 2      * 0:equal, 1:F1X4, 2:F3X4, 3:F61

    model = 0      * 0:one omega ratio for all branches  [FOR MODEL H0]
                   * 1:separate omega for each branch
                   * 2:user specified dN/dS ratios for branches [FOR MODELS H1-H3]

  NSsites = 0      *

    icode = 0      * 0:universal code

fix_kappa = 0      * 1:kappa fixed, 0:kappa to be estimated
    kappa = 2      * initial or fixed kappa

fix_omega = 0      * 1:omega fixed, 0:omega to be estimated
    omega = 0.2    * initial omega
```

NOTE: These hypotheses (H_0 → H_3) are actually specified in the four separate tree files!!!

```
*H0 in Table 3:
*model = 0
*(X02152Hom,U07178Sus,(M22585rab,((NM017025Rat,U13687Mus),
*(((AF070995C,(X04752Mus,U07177Rat)),(U95378Sus,U13680Hom)),(X53828OG1,
* U28410OG2)))));

*H1 in Table 3:
*model = 2
*(X02152Hom,U07178Sus,(M22585rab,((NM017025Rat,U13687Mus),(((AF070995C,
*(X04752Mus,U07177Rat)),(U95378Sus,U13680Hom))#1,(X53828OG1,U28410OG2))
* )));

*H2 in Table 3:
*model = 2
*  (X02152Hom,U07178Sus,(M22585rab,((NM017025Rat,U13687Mus),(((AF070995C
*  #1,(X04752Mus #1,U07177Rat #1)#1)#1,(U95378Sus #1,U13680Hom #1)
*  #1)#1,(X53828OG1,U28410OG2)))));

*H3 in Table 3:
*model = 2
*  (X02152Hom,U07178Sus,(M22585rab,((NM017025Rat,U13687Mus),(((AF070995C
*  #1,(X04752Mus #1,U07177Rat #1)#1)#1,(U95378Sus #1,U13680Hom #1)
*  #1)#1,(X53828OG1 #2,U28410OG2 #2)#2))));
```



H_0: $\omega_{A0} = \omega_{A1} = \omega_{C1} = \omega_{C0}$
H_1: $\omega_{A0} = \omega_{A1} = \omega_{C1} \neq \omega_{C0}$
H_2: $\omega_{A0} = \omega_{A1} \neq \omega_{C1} = \omega_{C0}$
H_3: $\omega_{A0} \neq \omega_{A1} \neq \omega_{C1} = \omega_{C0}$

exercise 3:

Complete this table **AND Interpret your findings**

Table E3: Parameter estimates under models of variable ω ratios among lineages and LRTs of their fit to the *Ldh-A* and *Ldh-C* gene family.

Models	ω_{A0}	ω_{A1}	ω_{C1}	ω_{C0}	ℓ	LRT
H_0: $\omega_{A0} = \omega_{A1} = \omega_{C1} = \omega_{C0}$	?	= $\omega_{A.0}$	= $\omega_{A.0}$	= $\omega_{A.0}$	?	na
H_1: $\omega_{A0} = \omega_{A1} = \omega_{C1} \neq \omega_{C0}$	?	= $\omega_{A.0}$	= $\omega_{A.0}$	?	?	?
H_2: $\omega_{A0} = \omega_{A1} \neq \omega_{C1} = \omega_{C0}$	?	= $\omega_{A.0}$	?	= $\omega_{C.1}$	?	**?**
H_3: $\omega_{A0} \neq \omega_{A1} \neq \omega_{C1} = \omega_{C0}$	?	?	?	= $\omega_{C.1}$	?	?

The topology and branch specific ω ratios are presented in Figure 5.
H_0 v H_1: df = 1
H_0 v H_2: df = 1
H_2 v H_3: df = 1

When you interpret your results, THINK about why the involved models are nested.

exercise 3 concept questions:

1. Can you explain the biological interpretation of all 4 models (hypotheses) of *Ldh* gene-family evolution?

2. Can you explain how these models are nested. Why is nesting a concern here? Do you understand the df for the relevant LRTs?

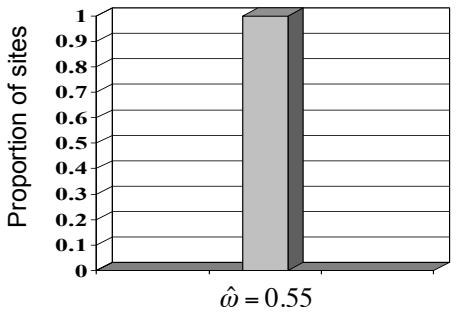
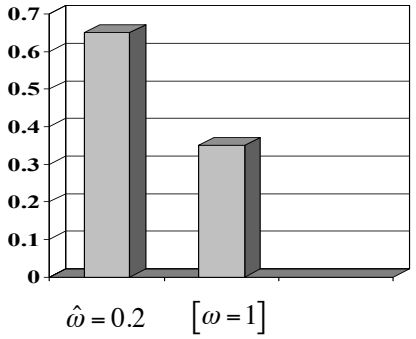
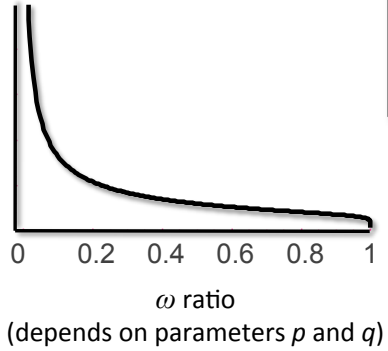
3. What evolutionary scenario is the best explanation of *Ldh* gene-family evolution?

4. Is there evidence of positive selection during the history of *Ldh* evolution? Are there any scenarios in which *Ldh* could have evolved by positive selection that would be undetectable by these LRTs?

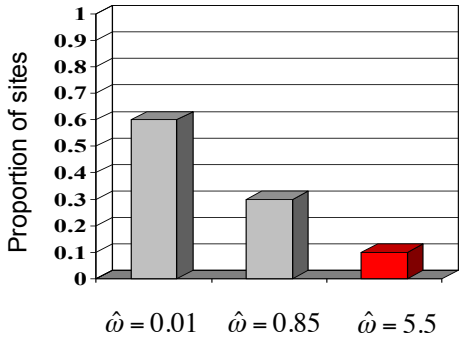
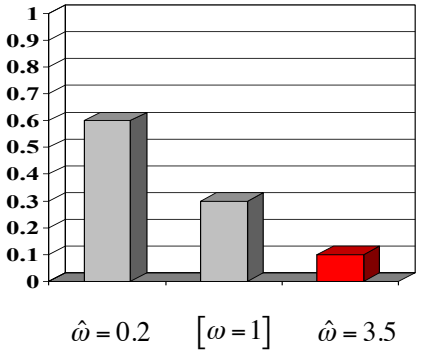
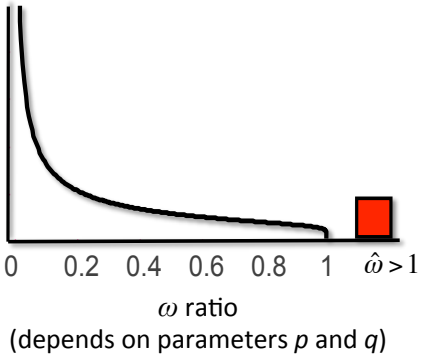
Exercise 4:

Testing for adaptive evolution in the *nef* gene of human HIV-2

exercise 4:

	M0	M1a	M7
H_0			

Now recommend a restricted version of M8 for the 3rd LRT (instead of M7)

	M3	M2a	M8
H_a			

LRT			
	1: M0 vs. M3 test for variable selection pressure among sites	2: M1a vs. M2a tests for sites subject to positive selection	3: M7 vs. M8 tests for sites subject to positive selection
	df = 4	df = 2	df = 2

```
      seqfile = seqfile.txt          * sequence data filename

    * treefile = treefile_M0.txt     * SET THIS for tree file with ML branch lengths under M0
    * treefile = treefile_M1.txt     * SET THIS for tree file with ML branch lengths under M1
    * treefile = treefile_M2.txt     * SET THIS for tree file with ML branch lengths under M2
    * treefile = treefile_M3.txt     * SET THIS for tree file with ML branch lengths under M3
    * treefile = treefile_M7.txt     * SET THIS for tree file with ML branch lengths under M7
    * treefile = treefile_M8.txt     * SET THIS for tree file with ML branch lengths under M8

      outfile = results.txt          * main result file name
        noisy = 9                    * lots of rubbish on the screen
      verbose = 1                    * detailed output
      runmode = 0                    * user defined tree
      seqtype = 1                    * codons
    CodonFreq = 2                    * F3X4 for codon ferquencies
        model = 0                    * one omega ratio for all branches

    * NSsites = 0                    * SET THIS for M0
    * NSsites = 1                    * SET THIS for M1
    * NSsites = 2                    * SET THIS for M2
    * NSsites = 3                    * SET THIS for M3
    * NSsites = 7                    * SET THIS for M7
    * NSsites = 8                    * SET THIS for M8

        icode = 0                    * universal code
    fix_kappa = 1                    * kappa fixed
      * kappa = 4.43491              * SET THIS to fix kappa at MLE under M0
      * kappa = 4.39117              * SET THIS to fix kappa at MLE under M1
      * kappa = 5.08964              * SET THIS to fix kappa at MLE under M2
      * kappa = 4.89033              * SET THIS to fix kappa at MLE under M3
      * kappa = 4.22750              * SET THIS to fix kappa at MLE under M7
      * kappa = 4.87827              * SET THIS to fix kappa at MLE under M8

    fix_omega = 0                    * omega to be estimated
        omega = 5                    * initial omega

      * ncatG = 3                    * SET THIS for 3 site categories under M3
      * ncatG = 10                   * SET THIS for 10 of site categories under M7 and M8

  fix_blength = 2                    * fixed branch lengths from tree file
```

These trees contain **pre-computed MLEs for branch lengths** to speed the analyses.

You will want to estimate all the branch lengths via ML when you analyze your own data!

Be careful: there is a lot to change in this codeml.ctl file for each model.

It is very easy to miss something, or make a mistake

The models will run quick, so it is also easy to check/fix any mistakes.

exercise 4:

Complete this table **AND Interpret your findings**

Table E4: Parameter estimates and likelihood scores under models of variable ω ratios among sites for HIV-2 *nef* genes.

Nested model pairs	d_N/d_S [b]	Parameter estimates [c]	PSS [d]	ℓ
M0: one-ratio (1) [a]	?	ω = ?	N.A.	?
M3: discrete (5)	?	p_0, = ?, p_1, = ?, (p_2 = ?) ω_0 = ?, ω_1 = ?, ω_2 = ?	? (?)	?
M1a: neutral (2)	?	p_0 = ?, (p_1 = ?) ω_0 = ?, (ω_1 = 1)	N.A.	?
M2a: selection (4)	?	p_0= ?, p_1 = ?, (p_2 = ?) ω_0 = ?, (ω_1 = 1), ω_2 = ?	? (?)	?
M7: beta (2)	?	p = ?, q = ?	N.A.	?
M8: beta&ω (4)	?	p_0 = ? (p_1 = ?) p = ?, q = ?, ω = ?	? (?)	?

[a] The number after the model code, in parentheses, is the number of free parameters in the ω distribution.
[b] This d_N/d_S ratio is an average over all sites in the HIV-2 *nef* gene alignment.
[c] Parameters in parentheses are not free parameters.
[d] PSS is the number of positive selection sites (NEB). The first number is the PSS with posterior probabilities > 50%. The second number (in parentheses) is the PSS with posterior probabilities > 95%.

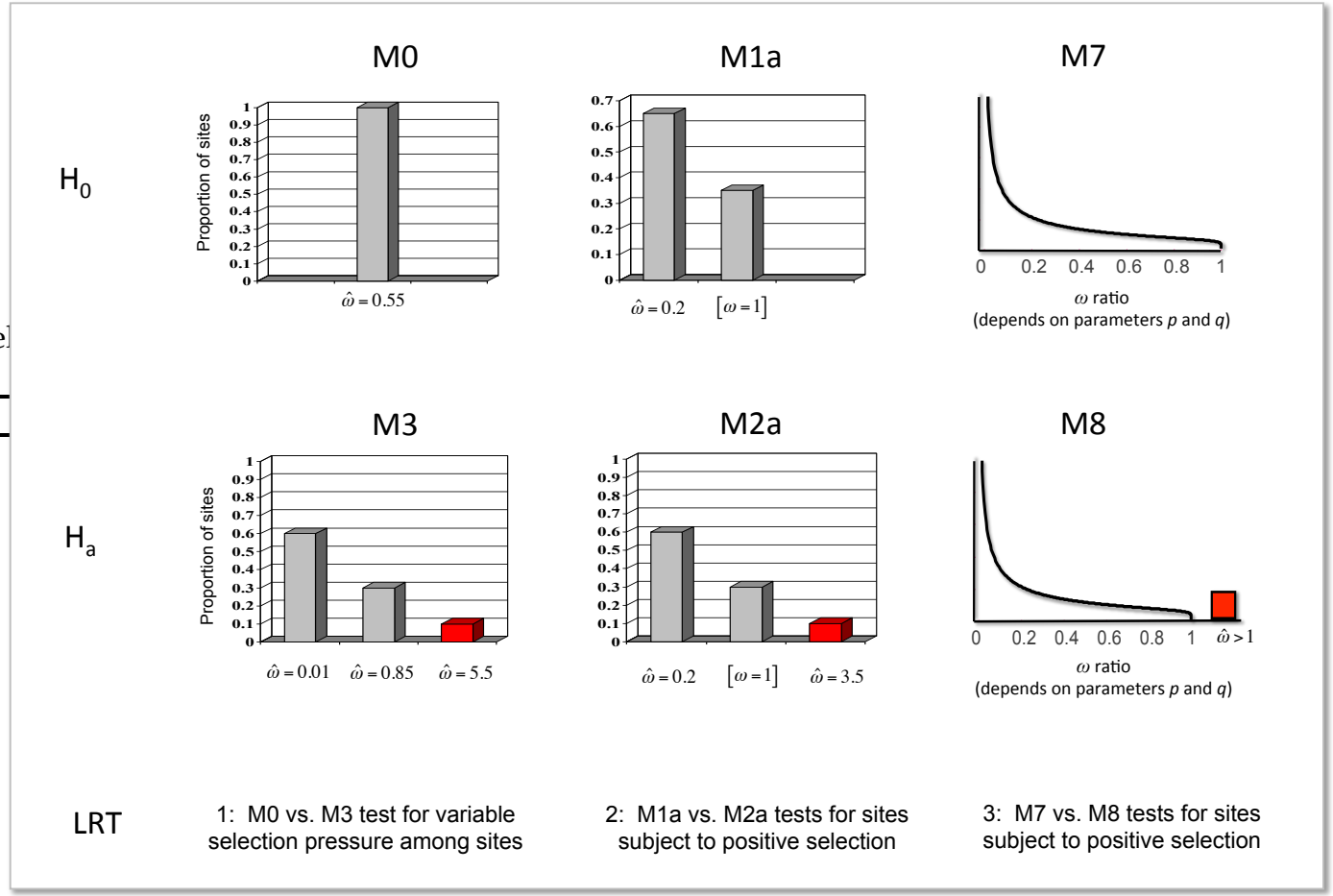
exercise 4:

Complete this table **AND Interpret**

Table E4: Parameter estimates and likelihood scores under model
sites for HIV-2 *nef* genes.

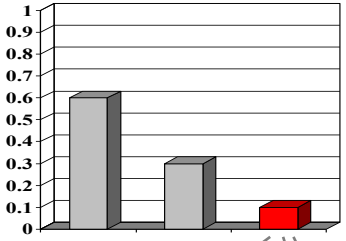
Nested model pairs	d_N/d_S [b]	Parameter estimates [c]
M0: one-ratio (1) [a]	?	ω = ?
M3: discrete (5)	?	p_0, = ?, p_1, = ?, (p_2 = ?) ω_0 = ?, ω_1 = ?, ω_2 = ?
M1a: neutral (2)	?	p_0 = ?, (p_1 = ?) ω_0 = ?, (ω_1 = 1)
M2a: selection (4)	?	p_0= ?, p_1 = ?, (p_2 = ?) ω_0 = ?, (ω_1 = 1), ω_2 = ?
M7: beta (2)	?	p = ?, q = ?
M8: beta&ω (4)	?	p_0 = ? (p_1 = ?) p = ?, q = ?, ω = ?

[a] The number after the model code, in parentheses, is the number of free parameters in the ω distribution.
[b] This d_N/d_S ratio is an average over all sites in the HIV-2 *nef* gene alignment.
[c] Parameters in parentheses are not free parameters.
[d] PSS is the number of positive selection sites (NEB). The first number is the PSS with posterior probabilities > 50%. The second number (in parentheses) is the PSS with posterior probabilities > 95%.

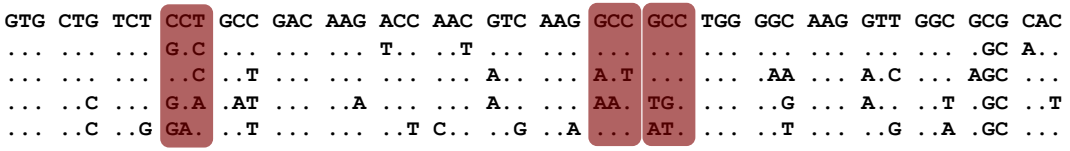


Concept map for tasks 1-3…

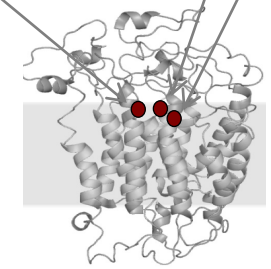
model:
5% have $\omega > 1$



Bayes' rule:
site 4, 12 & 13

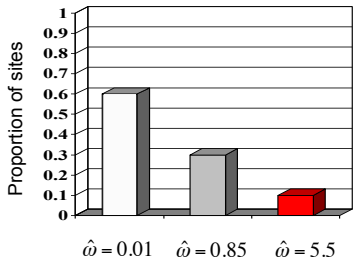


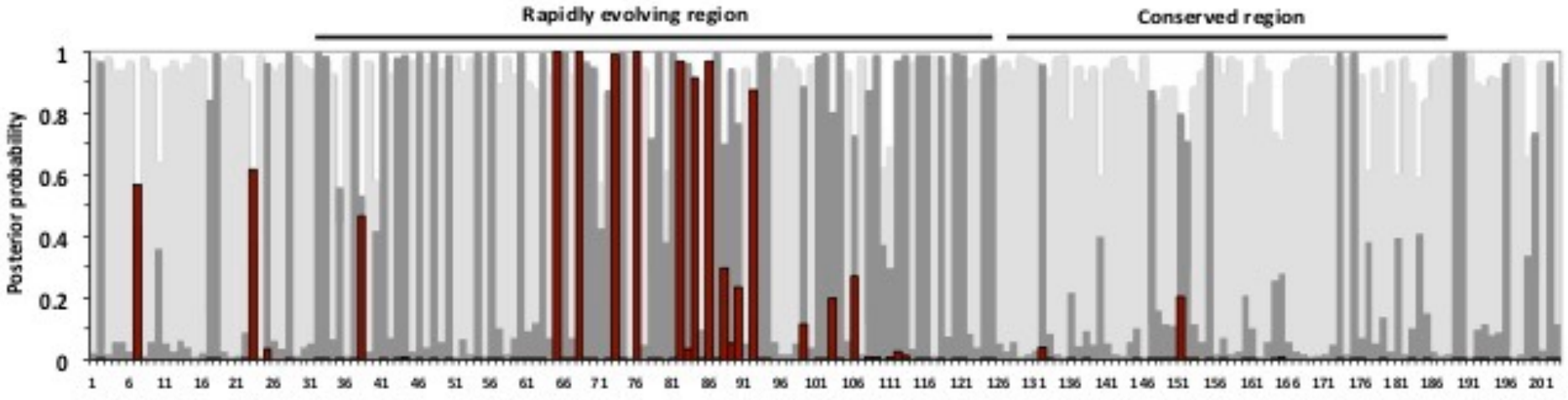
structure:
sites are in contact



1. Fit model to data → MLEs
2. Test hypotheses via Null and alternative models for ω
3. Predict which sites have $\omega>1$
4. Interpret results in known biological context

exercise 4: use the "**rst file**" for model **M3** to produce a plot like this for the *nef* gene





NOTE: This is **NOT** the distribution for the *nef* gene

exercise 4 concept questions:

Try to synthesize all your results and attempt a biological interpretation of the sort that you would want to publish within an actual research paper. The following two general questions should help get you going. I strongly encourage you to do this last step in collaboration with other workshop students; talk it through!

1. What *biological conclusions* are well-supported by these data?

2. What aspects of the results can you interpret according your prior biological knowledge of this, or similar, systems?